- 3 -

<u>IN THE CLAIMS</u>

Amended claims follow:

1.   (Original) A computer program product for controlling a computer, said computer program product comprising:

        (i) scan request receiving logic operable to receive a request to perform an on-access malware scan upon a computer file to which access is to be made;

        (ii) scan dividing logic operable to divide said on-access malware scan into a plurality of tasks;

        (iii) task issuing logic operable to issue said plurality of tasks to be performed by a plurality of different computers; and

        (iv) result collating logic operable to collate a plurality of task results corresponding to said plurality of tasks and received from said plurality of different computer to form a scan result corresponding to said on-access malware scan.

2.   (Original) A computer program product as claimed in claim 1, wherein said scan dividing logic divides said computer file into a plurality of component computer files to be separately scanned as said plurality of tasks.

3.   (Original) A computer program product as claimed in claim 2, wherein said computer file contains one or more embedded computer files which are divided out as component computer files.

4.   (Original) A computer program product as claimed in claim 3, wherein said computer file is one of the following computer file types: OLE2, ZIP, CAB, ARJ, RAR, ACE, JAR, ARC, LHA, LZH, ICE and StuffIt.

5.   (Original) A computer program product as claimed in claim 1, wherein said scan dividing logic divides said on-access malware scan into a plurality of on-access malware scans for identifying different properties of said computer file, said plurality of on-access

malware scans being separately performed as said plurality of tasks.

6.   (Original) A computer program product as claimed in claim 5, wherein said plurality of tasks each seek to identify different portions of one of a cryptographic analysis and an emulation analysis.

7.   (Original) A computer program product as claimed in claim 1, wherein said on-access malware scan of said computer file seeks to identify one or more of:

        (i) a computer virus;

        (ii) a Trojan computer program;

        (iii) a worm computer program;

        (iv) a banned computer program; and

        (v) an e-mail containing banned content.

8.   (Original) A computer program product as claimed in claim 1, wherein one or more of said tasks are liner divided into sub-tasks.

9.   (Currently Amended) A computer program product as claimed in claim 1, wherein a task is selected to be issued to a different computer in dependence upon one [of]or more of:

        (i) a measure of available processing resources at said different computer;

        (ii) a measure of communication channel bandwidth to said different computer;

        (iii) a measure of task complexity of said task to be issued; and

        (iv) a measure of processor utilization of said different computer.

10.   (Original) A computer program product as claimed in claim 1, wherein said scan dividing logic does not divide said on-access malware scan if said on-access malware scan is detected as having a complexity below a predetermined threshold level.

11.   (Original) A computer program product as claimed in claim 10, wherein said complexity is determined as a function of one or more of:

- 5 -

(i) a file type of said computer file;

(ii) whether said computer file contains any embedded computer files; (iii) a level of nesting of embedded files within said computer file;

(iv) an initial scanning attempt of said computer file taking longer than a predetermined time; and

(v) processor utilization of a computer initiating said request.

12.   (Original) A computer program product as claimed in claim 1, wherein said result collating logic terminates any outstanding tasks if a task result is received indicating detection of malware within said computer file.

13.   (Currently Amended) A computer program product for controlling a computer, said computer program product comprising:

(i) task receiving logic operable to receive a request to perform a malware scanning task that is part of an on-access malware scan of a computer file requested by another computer;

(ii) scanning logic operable to perform said malware scanning task; and

(iii) result returning logic operable to return a result of said malware scanning task;

wherein said malware scanning task is one of a plurality of malware scanning tasks that are each part of said on-access malware scan;

wherein a plurality of malware scanning task results corresponding to said plurality of malware scanning tasks are collated to form a scan result corresponding to said on-access malware scan .

14.   (Original) A computer program product as claimed in claim 13, wherein said computer file is divided into a plurality of component computer files to be separately scanned as separate malware scanning tasks.

15.   (Currently Amended) A computer program product as claimed in claim 13, wherein said on-access malware scan of said computer file is divided into [a]said

- 6 -

plurality of on-access malware scanning tasks for identifying different properties of said computer file, said plurality of on-access malware scanning tasks being separately performed.

16.   (Currently Amended) A method of performing an on-access malware scan of a computer file, said method comprising the steps of:

(i) receiving a request to perform an on-access malware scan upon a computer file to which access is to be made;

(ii) dividing said on-access malware scan into a plurality of tasks;

(iii) issuing said plurality of tasks to be performed by a plurality of different computers; and

(iv) collating a plurality of task results corresponding to said plurality of tasks and received from said plurality of different computer to form a scan result corresponding to said on-access malware scan. ~~The easiest way to split is when a subtask is a separate scannable object (another file embedded into OLE or part of an archive).~~

17.   (Original) A method as claimed in claim 1, wherein said computer file is divided into a plurality of component computer files to be separately scanned as said plurality of tasks.

18.   (Original) A method as claimed in claim 17, wherein said computer file contains one or more embedded computer files which are divided out as component computer files.

19.   (Original) A method as claimed in claim 18, wherein said computer file is one of the following computer file types: OLE2, ZIP, CAB, ARJ, RAR, ACE, JAR, ARC, LHA, LZH, ICE and StuffIt.

20.   (Original) A method as claimed in claim 16, wherein said on-access malware scan is divided into a plurality of on-access malware scans for identifying different properties of said computer file, said plurality of on-access malware scans being separately

- 7 -

performed as said plurality of tasks.

21. (Original) A method as claimed in claim 20, wherein said plurality of tasks each seek to identify different portions of one of a cryptographic analysis and an emulation analysis.

22. (Original) A method as claimed in claim 16, wherein said on-access malware scan of said computer file seeks to identify one or more of:

 (i) a computer virus;

 (ii) a Trojan computer program;

 (iii) a worm computer program;

 (iv) a banned computer program; and

 (v) an e-mail containing banned content.

23. (Original) A method as claimed in claim 16, wherein one or more of said tasks are further divided into sub-tasks.

24. (Currently Amended) A method as claimed in claim 16, wherein a task is selected to be issued to a different computer in dependence upon one [of]or more of:

 (i) a measure of available processing resources at said different computer;

 (ii) a measure of communication channel bandwidth to said different computer;

 (iii) a measure of task complexity of said task to be issued; and

 (iv) a measure of processor utilization of said different computer.

25. (Original) A method as claimed in claim 16, wherein said on-access malware scan is not divided if said on-access malware scan is detected as having a complexity below a predetermined threshold level.

26. (Original) A method as claimed in claim 25, wherein said complexity is determined as a function of one or more of:

 (i) a file type of said computer file;

me

- 8 -

(ii) whether said computer file contains any embedded computer files;

(iii) a level of nesting of embedded files within said computer file;

(iv) an initial scanning attempt of said computer file taking longer than a predetermined time; and

(v) processor utilization of a computer initiating said request.

27.   (Original) A method as claimed in claim 16, wherein any outstanding tasks are terminated if a task result is received indicating detection of malware within said computer file.

28.   (Currently Amended) A method of on-access malware scanning, said method comprising the steps of:

(i) receiving a request to perform a malware scanning task that is part of an on-access malware scan of a computer file requested by another computer;

(ii) performing said malware scanning task; and

(iii) returning a result of said malware scanning task;

wherein said malware scanning task is one of a plurality of malware scanning tasks that are each part of said on-access malware scan;

wherein a plurality of malware scanning task results corresponding to said plurality of malware scanning tasks are collated to form a scan result corresponding to said on-access malware scan.

29.   (Original) A method as claimed in claim 28, wherein said computer file is divided into a plurality of component computer files to be separately scanned as separate malware scanning tasks.

30.   (Currently Amended) A method claimed in claim 28, wherein said on-access malware scan of said computer file is divided into [a]said plurality of on-access malware scanning tasks for identifying different properties of said computer file, said plurality of on-access malware scanning tasks being separately performed.

- 9 -

31.   (Original) Apparatus for performing an on-access malware scan of a computer
file, said apparatus comprising:

(i) a scan request receiver operable to receive a request to perform an on-access
malware scan upon a computer file to which access is to be made;

(ii) a scan divider operable to divide said on-access malware scan into a plurality
of tasks;

(iii) a task issuer operable to issue said plurality of tasks to be performed by a
plurality of different computers; and

(iv) a result collator operable to collate a plurality of task results corresponding to
said plurality of tasks and received from said plurality of different computer to form a
scan result corresponding to said on-access malware scan.

32.   (Original) Apparatus as claimed in claim 31, wherein said scan divider divides
said computer file into a plurality of component computer files to be separately scanned
as said plurality of tasks.

33.   (Original) Apparatus as claimed in claim 32, wherein said computer file contains
one or more embedded computer files which are divided out as component computer
files.

34.   (Original) Apparatus as claimed in claim 33, wherein said computer file is one of
the following computer file types: OLE2, ZIP, CAB, ARJ, RAR, ACE, JAR, ARC, LHA,
LZH, ICE and StuffIt.

35.   (Original) Apparatus as claimed in claim 31, wherein said scan divider divides
said on-access malware scan into a plurality of on-access malware scans for identifying
different properties of said computer file, said plurality of on-access malware scans being
separately performed as said plurality of tasks.

36.   (Original) Apparatus as claimed in claim 35, wherein said plurality of tasks each
seek to identify different portions of one of a cryptographic analysis and an emulation

- 10 -

analysis.

37.   (Original) Apparatus as claimed in claim 31, wherein said on-access malware scan of said computer file seeks to identify one or more of:

      (i) a computer virus;

      (ii) a Trojan computer program;

      (iii) a worm computer program;

      (iv) a banned computer program; and an e-mail containing banned content.

38.   (Original) Apparatus as claimed in claim 31, wherein one or more of said tasks are further divided into sub-tasks.

39.   (Currently Amended) Apparatus as claimed in claim 31, wherein a task is selected to be issued to a different computer in dependence upon one [of]or more of:

      (i) a measure of available processing resources at said different computer;

      (ii) a measure of communication channel bandwidth to said different computer;

      (iii) a measure of task complexity of said task to be issued; and

      (iv) a measure of processor utilization of said different computer.

40.   (Original) Apparatus as claimed in claim 31, wherein said scan divider does not divide said on-access malware scan if said on-access malware scan is detected as having a complexity below a predetermined threshold level.

41.   (Original) Apparatus as claimed in claim 40, wherein said complexity is determined as a function of one or more of:

      (i) a file type of said computer file;

      (ii) whether said computer file contains any embedded computer files;

      (iii) a level of nesting of embedded files within said computer file;

      (v) an initial scanning attempt of said computer file taking longer than a predetermined time; and

- 11 -

(vi) processor utilization of a computer initiating said request.

42.   (Original) Apparatus as claimed in claim 31, wherein said result collator terminates any outstanding tasks if a task result is received indicating detection of malware within said computer file.

43.   (Currently Amended) Apparatus for performing an on-access malware scan of a computer file, said apparatus comprising:

(i) a task receiver operable to receive a request to perform a malware scanning task that is part of an on-access malware scan of a computer file requested by another computer;

(ii) a scanner operable to perform said malware scanning task; and

(iii) a result returner operable to return a result of said malware scanning task;

wherein said malware scanning task is one of a plurality of malware scanning tasks that are each part of said on-access malware scan;

wherein a plurality of malware scanning task results corresponding to said plurality of malware scanning tasks are collated to form a scan result corresponding to said on-access malware scan.

44.   (Original) Apparatus as claimed in claim 43, wherein said computer file is divided into a plurality of component computer files to be separately scanned as separate malware scanning tasks.

45.   (Currently Amended) Apparatus as claimed in claim 43, wherein said on-access malware scan of said computer file is divided into [a]said plurality of on-access malware scanning tasks for identifying different properties of said computer file, said plurality of on-access malware scanning tasks being separately performed.

46.   (New) A computer program product as claimed in claim 1, wherein said scan dividing logic divides said on-access malware scan in response to a complexity metric

exceeding a predetermined threshold, where the complexity metric is dependent on at least one parameter.

47.   (New) A computer program product as claimed in claim 46, wherein the parameter includes at least one of a computer file type, a level of nesting of embedded computer files, an initial attempt to scan said computer file which exceeded a predetermined time, and a level of utilization of a local processor.

48.   (New) A computer program product as claimed in claim 46, wherein the complexity metric is dependent on a plurality of parameters including a computer file type, a level of nesting of embedded computer files, an initial attempt to scan said computer file which exceeded a predetermined time, and a level of utilization of a local processor.

49.   (New) A computer program product as claimed in claim 46, wherein an amount said complexity metric exceeds said predetermined threshold determines a number of tasks into which said on-access malware scan is divided.

50.   (New) A computer program product as claimed in claim 1, wherein if one of said plurality of task results indicates that malware has been detected in said computer file, remaining tasks in said plurality of tasks that are pending are terminated.

51.   (New) A computer program product as claimed in claim 1, wherein said plurality of tasks are distributed among said plurality of different computers via a network.